



Finite-state discrete-time Markov chain models of gene regulatory networks

Vladimir Skornyakov, Maria Skornyakova, Antonina Shurygina, et al.

bioRxiv first posted online June 23, 2014

Access the most recent version at doi: <http://dx.doi.org/10.1101/006361>

Creative
Commons
License

The copyright holder for this preprint is the author/funder. It is made available under a [CC-BY-NC 4.0 International license](#).

Finite-state discrete-time Markov chain models of gene regulatory networks

Skornyakov V.P., Skornyakova M.V., Shurygina A.V. , and
Skornyakov P.V.

Kogan Research Institute of Neurocybernetics, Southern Federal
University, Rostov-on-Don, Russia

June 23, 2014

Abstract

In this study Markov chain models of gene regulatory networks (GRN) are developed. These models gives the ability to apply the well known theory and tools of Markov chains to GRN analysis. We introduce a new kind of the finite graph of the interactions called the combinatorial net that formally represent a GRN and the transition graphs constructed from interaction graphs. System dynamics are defined as a random walk on the transition graph that is some Markovian chain. A novel concurrent updating scheme (evolution rule) is developed to determine transitions in a transition graph. Our scheme is based on the firing of a random set of non-steady state vertices of a combinatorial net. We demonstrate that this novel scheme gives an advance in the modeling of the asynchronicity. Also we proof the theorem that the combinatorial nets with this updating scheme can asynchronously compute a maximal independent sets of graphs. As proof of concept, we present here a number of simple combinatorial models: a discrete model of auto-repression, a bi-stable switch, the Elowitz repressilator, a self-activation and show that this models exhibit well known properties.

1 Introduction

Efforts to study gene expression regulation networks has led to a detailed description many of them, and many more are to be identified in the near future. Therefore there is a need to develop methods of computational and theoretical analysis of GRNs. One of the most promising directions is to reduce the problem to the study of Markov chains, generated in some way from the GRN [1, 2, 3, 4, 5]. Usually Boolean networks [6] are used as a formal representation of GRN. Classification of process states, the study of long-term behavior [7], and development of optimal strategies for therapeutic intervention [8, 9, 10, 7, 11, 12, 13, 14, 15] provide good examples of this approach [16]. In contrast to the Boolean network, the Hopfield networks are defined using arithmetic operations [17]. It is a well-developed branch of science which deals with stochastic processes of asynchronous state switching as a result of

interactions. As such they are similar to Boolean networks. A Hopfield like formalism also leads to the definition of the Markov chain. In the Hopfield network area essential results were obtained in the study of various update schemes [18], network oscillations [19], solving of combinatorial optimization problems [20, 21, 22, 23, 19, 24, 25, 26], and estimating the rate of the convergence and many others. This makes it valuable to study the possibility of using Hopfield like networks for the construction of Markov chains from GRNs and other interactions graphs. We consider a GRN as a kind of interaction graph. Interaction (regulatory) graphs are emerged in various fields of the life science [27]. Nowadays, their transition graphs are often used to analyze properties of interactions (regulations). One promising way to understand the nature of the regulation or interactions represented by interaction graphs is to analyze the Markovian chain associated with their transition graphs.

2 Method

The proposed method may be viewed as a version of the Hopfield like network [17] where groups of randomly selected *unstable units* are updated in parallel [18].

2.1 The interactions graph and non steady state vertices

Let $G = (V, E)$ be a directed graph, where V is set of vertices and E is set of edges. Let $B = \{0, 1\}$ be a set of vertex states. We say that the vertex is *active* if the state of this vertex is equal to "1", otherwise we say that vertex is *inactive*. The map function $M: V \rightarrow B$ give us the state of each vertex. If some vertex $v \in V$, then $M(v)$ is the state of the vertex v that correspond to map M . $M(v) = 1$ is equivalent to the vertex v is active under map M . $M(v) = 0$ is equivalent to the vertex v is inactive. The weight function $W: E \rightarrow R$ gives the value of each edge of the graph G , which represent the power of interactions. If $e = (u, v)$ than we say that u is a direct ancestor of v and v is a direct descendant of u . The *influence* on v under the map M is defined as the sum of weights of edges from all direct active ancestors of vertex v . The influence on v under the map M is denoted by $I(v, M)$ (also called "the local field or the net input"). That is

$$I(v, M) = \sum_{\{u | (u, v) \in E\}} W((u, v)) \cdot M(u) \quad (1)$$

This influence is determined by map function M and weight function W . Only if the weight function W is assumed to be constant over time then we say that influence $I(v, M)$ is influence of map M on vertex v . If $I(v, M) \geq 0$ we say that map M activate vertex v otherwise we say that map M repress vertex v . Now we can give most important definition of vertex steady state under the map M . Let v be the active under map M , if map M activate v then the state of v is a steady state under the map M , else the state of v is a non steady state under the map M . Let v be inactive under map M , if map M inactivate v then the state of v is a steady state under the map M , else the state of v is a non steady state under the map M . If all vertices are in steady state under the map

M we say that map M is steady map. The *forced* state of the vertex v under map M is defined as follows:

$$F(v, M) = \begin{cases} 0, & \text{if } I(v, M) < 0 ; \\ 1, & \text{if } I(v, M) \geq 0 . \end{cases} \quad (2)$$

By definition, if forced state and current state of v are the same then state of the vertex v under the map M is steady. Next equation give us the formal definition:

$$\text{steady}(v, M) = \begin{cases} 0, & \text{if } F(v, M) = M(v) ; \\ 1, & \text{if } F(v, M) \neq M(v) . \end{cases} \quad (3)$$

2.2 The Random Set update rule

Now we consider a stochastic process $\{Y_j, j = 1, 2, 3, \dots\}$ that takes on the set of maps of some interaction graph G , where Y_j denote the map of G at time period j . At each time period j for each non steady-state vertex v_i under the map Y_j we change the current vertex state to a forced state with the probability p_i , and the current state stays unchanged with the probability $1 - p_i$. Let $S = \{v_1, v_2, \dots, v_n\}$ be a set of all non steady-state vertices at the time period j . Vertices chosen to change their state in one-step transition form a random set $X \subseteq S$. To make the new map that direct accessible from current map Y_j all vertices from X simultaneously change their state whereas another vertices stay unchanged. Let $P = \{p_1, p_2, \dots, p_n\}$ be some vector of numbers such that $0 \leq p_i \leq 1$, we refer to p_i as the probability of the state changing (firing) of the non steady-state vertex v_i . For any $X \subseteq S$ let $\mathbf{1}_X: S \rightarrow B$ be indicator function such that $\mathbf{1}_X(v_i) = 1$ if $v_i \in X$, otherwise $\mathbf{1}_X(v_i) = 0$. Let $\mathbf{P}_X: S \rightarrow [0.0, 1.0]$ denote the function such that $\mathbf{P}_X(v_i) = p_i$ if $\mathbf{1}_X(v_i) = 1$, otherwise $\mathbf{P}_X(v_i) = 1 - p_i$. Hence we assume that each v_i acts independently to make random set X then the production of \mathbf{P}_X give us p_X that is the firing probability of the random set X

$$p_X = \prod_{v \in S} \mathbf{P}_X(v) \quad (4)$$

Evidently,

$$\sum_{X \subseteq S} \prod_{v \in S} \mathbf{P}_X(v) = 1 \quad (5)$$

Now we can apply this definition of Random Set update rule and its probabilities to define the transitions graph of the combinatorial net models.

2.3 The transitions graph of the interactions graph

Let S be the set of non steady state vertices under map $M = Y_j$. S represent a full set of vertices each of which can flip to a forced state at next $j+1$ time step. In the combinatorial model steady state vertices can not flip. To construct a transition graph we should define the full set of maps M_1, M_2, \dots that are direct reachable from the map M . Each pair (M, M_i) will correspond to one edge in transition graph. What is the set of maps M_1, M_2, \dots , which can be direct (by one-step transition) reached from map M ? To represent the independence and the asynchronicity we assume that any random set of non steady state vertices

$X \subseteq S$ can produce next map from current map. Let M_X be map such that $M_X(v) = F(v, M)$ if $v \in X$ and $M_X(v) = M(v)$ if $v \notin X$, then we say that M_X produced by X from M . That is, the random set X of non steady state vertices produce the map M_X from a map M . The weight of edge from map M to a map M_X are given by the probability defined by equation (4).

2.4 The random walk network dynamics

We suppose that whenever the process is in state M , there is a probability p_X such that at next step it will be in state M_X . This probability is defined for each random set X of non-steady state vertices of map M .

3 Random Set update rules are more general

It is well known that asynchronous and Random Set update rules are equivalent in the sense of global stable states [28]. But in the sense of the reachability of one state from another they are not equivalent. Figure 1 show the Mace combinatorial model that illustrate this fact. The vertex e provides constant level of repression for vertex c , that is equal -2. Let the vertex d of the Mace will be active at the start. Then it can activate both middle vertices a and b . Due to repression, the vertex c of the Mace can be activated only if both middle vertices a and b will be active simultaneously. Asynchronous (one at a time) updating exclude simultaneous activation of these vertices, but Random Set update rule do not. Synchronous update rule do not exclude simultaneous activation of a and b , but it make the system deterministic. The Random Set update rule is more general than both synchronous and asynchronous update rules, because it allow all possible paths of a system evolution. Therefore, transition graphs of both synchronous and asynchronous update rules are subgraphs of Random Set update graph.

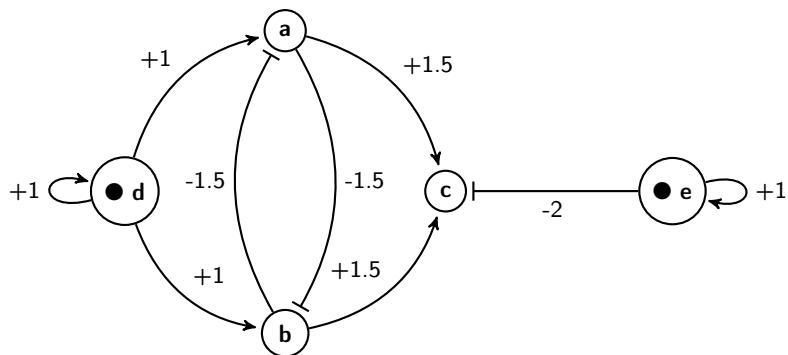


Figure 1: The Mace

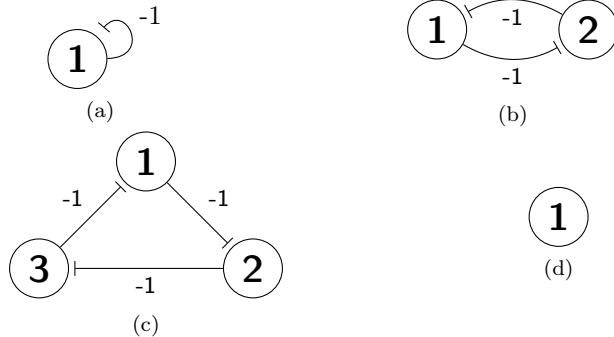


Figure 2: The interaction graph of: (a) The Autorepression model; (b) The Bistable Switch model; (c) The Elowitz repressilator model; (d) The self-activation model.

4 Examples of combinatorial models

Next we use described above method to develop models of some important graphs of repressive interactions of self activating nodes and prove their main properties. Such models we call combinatorial models.

4.1 The combinatorial model of an Auto-repression

A negative auto-regulation or an auto-repression occurs when products of some gene represses its own gene. This form of a simple regulation serve as a basic building block of most important transcription networks [27, 29]. An Auto-repression can produce oscillations. For example embryonic stem cells fluctuate between high and low Nanog expression and the Nanog activity is auto-repressive [30]. Our model of an auto-repression shown in Figure 3a and Figure 2a also exhibit stochastic oscillating behavior. Figure 2a present a graph of interactions $G = (V, E)$ of the Auto-repression model. The set V contains a single vertex v_1 . And the set E consist of a single edge $e_1 = (v_1, v_1)$ from vertex v_1 to itself. The weight of e_1 is equal to -1. Let the M_0 be a starting map and $M_0(v_1) = 0$, i.e. v_1 is inactive under the M_0 . Therefore, influence of M_0 to vertex v_1 equals -1, $I(v_1, M_0) = -1$. By equation [2], $F(v_1, M_0) = 0$, then state of the vertex v_1 is non steady under the map M_0 . Let $p_1 = 1/2$ then the state of the vertex v_1 could be changed to active state with probability $1/2$ and could stay unchanged with probability $1/2$. The another state of the model are also non steady. Therefore, there is only non steady states in the model. Thus, it will oscillate infinitely between 0 and 1. Figure 3a show full transition graph of the auto-repression model.

4.2 The combinatorial model of a Bi-stable Switch

A Bi-stable Switch is bi-stable gene regulatory network that constructed from a two mutually repressive genes [31]. They are widely common in nature and most used in synthetic biology [32, 33]. ODEs used to construct their mathematical models are convenient way for analyzing in detail some small circuits.

Here we develop techniques that can be used to construct models of large networks of bis-table switches and to prove some their important properties. Thus, we use probabilistic coarse-scale modeling approach [34] instead of fine-scale ODE modeling. Our model of Bis-table Switch shown in Fig 2b and Fig 3b exhibit two steady states. Figure 2b present a graph of interactions $G = (V, E)$ of the Bi-stable Switch model. The set $V = \{v_1, v_2\}$ contains two vertices and E contains two edges, which weights are -1. Probabilities of firing non steady state vertices let be 1/2. Figure 3b present the transition graph of the model. Maps 00 and 11 are non steady, whereas maps 01 and 10 are steady. To show that a map M is steady we must show that each vertex v is steady under map M . To show that a vertex v is steady under map M we must compute value of influence $F(v, M)$ of map M on vertex v and compare it to the state v under map M , i.e. $M(v)$. Now we will show that map 01 is steady.

4.3 The combinatorial model of the Elowitz Repressilator

Elowitz repressilator consists of three genes [35]. Each of these genes are constitutively expressed. The first gene inhibits the transcription of the second gene, whose protein product in turn inhibits the expression of a third gene, and finally, third gene inhibits first gene expression, completing the cycle. Such a negative feedback loop lead to oscillations. The combinatorial model of Elowitz repressilator produce oscillations and consists of three vertices and three edge, which weights is equal to -1. Figure 2c present a graph of interactions $G = (V, E)$ of the Elowitz repressilator model, where the set $V = \{v_1, v_2, v_3\}$ contains three vertices and $E = \{(v_1, v_2), (v_2, v_3), (v_3, v_1)\}$ contains three edges. Oscillations are produced by circle $\{1,3,2,6,4,5\}$ in Figure 3c. Decimal, binary, and graphical representations of the state space of the Elowitz repressilator are shown in Table 1.

4.4 The combinatorial model of the self-activation

A constitutively expressed gene represent an example of a self-activation. Such gene do not require any interaction to be active. A combinatorial model of the self-activation consist of one vertex and no edges. In any case the influence on it equals 0 since there are no other vertices. Therefore a forced state of the vertex equals 1. Thus 1 is steady state and 0 is non steady state. Vertex starting in steady state will stay in it infinitely. Vertex starting in non steady state with probability p flip to steady state and with probability $1-p$ stay in non steady state. The amount of time periods T which the vertex spent in non steady state is the random variable. The distribution of this random variable is the shifted geometric distribution with parameter p . Figures 2d and 3d represents the combinatorial model of a self-activation.

$$\Pr(T = k) = p(1 - p)^{k-1}$$

$$\mathbb{E}[T] = 1/(1 - p)$$

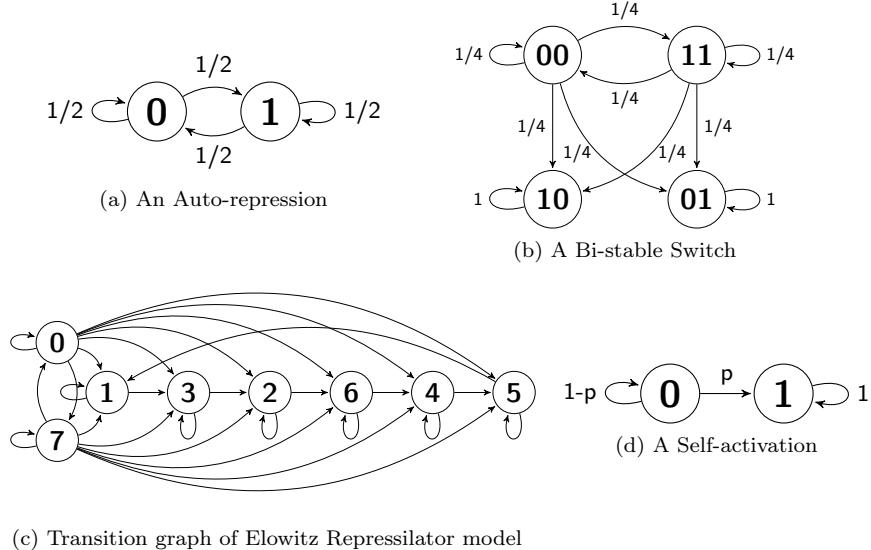


Figure 3: The Transition Graphs of : (a) The Autorepression model; (b) The Bistable Switch model; (c) The Elowitz repressilator model; (d) The self-activation model.

5 The network of bi-stable switches

An independent set (IS) in a graph is a set of vertices no two of which are adjacent . An independent set is called maximal(MIS) if there is no independent set that it contains properly. Hopfield network whose stable states are exactly maximal independent sets was developed by Shrivastava [36]. An independent set in a graph is a clique in the complement graph and vice versa. Thus, cliques can be used to find or to enumerate MISs [20, 21]. Finding independent sets (or cliques) has applications in various fields [37]. Combinatorial nets can be used to compute maximal independent sets of graphs in a *distributed self-organization* fashion. A stable states of *a network of bi-stable switches derived from a graph* are exactly maximal independent sets of its underlying graph. Now we consider simple graph H , that is, graph without directed, multiple, or weighted edges, and without self loops. Let $C(H)$ denote the graph obtained from H by deleting an each undirected edge (u, v) of H and adding instead of this edge new two directed edges (u, v) and (v, u) . The set of vertices of $C(H)$ let be the same as the set of vertices of H . Let -1 be a weight of each edge of $C(H)$. The Bi-stable Switch can be seen as combinatorial net derived from simple graph of order 2.

For example Figure 4b demonstrates the network derived from the graph shown in Figure 4a. The first switch is formed by the subgraph induced by the $\{1,2\}$ set of vertices of the $C(H)$ network . The second switch is formed by the $\{2,3\}$ set of vertices. The vertex 2 is common one of these switches, therefore they interact by means of this vertex. Each edge of an underlying graph correspond to a switch in a derived network. If two incident edges share a common vertex then the corresponding switches interact because this vertex have the same state in both of them. We refer to $C(H)$ as the *derived network* of bi-stable switches, and we refer to H as *the underlying graph*.

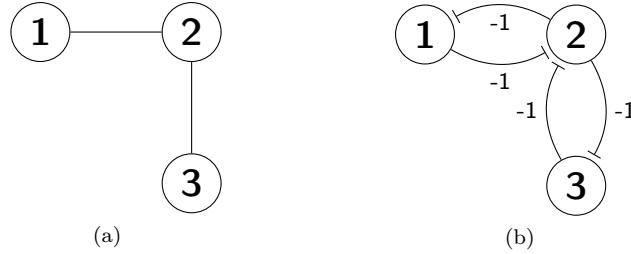


Figure 4: The network $C(H)$ derived from a graph H : (a) The graph H ; (b) The network $C(H)$ derived from the graph H ;

Lemma 1. A steady state map M of a combinatorial net that derived from a simple graph does not have two adjacent active vertices u and v .

Proof. Assume by contradiction that u and v are adjacent active vertices, then $I(u, M) < 0$ and the forced state of the u is inactive. If a state is steady and the forced state is inactive then the state of the vertex is inactive, whereas by conditions of the lemma u is active.

Lemma 2. *If a map M of a combinatorial net that derived from a simple graph is steady and some vertex is inactive then there is at least one active vertex adjacent to it.*

Proof. Assume by contradiction, that there are no active adjacent vertices of v , then influence $I(v, M) = 0$ and then forced state of v is active. Hence the map M is steady state, forced state of vertex is equal to current state, then state of the vertex is active in contradiction to conditions of the lemma.

Lemma 3. Under steady state map M of the combinatorial net $C(H)$ derived from simple graph H the set of all active vertices is maximal independent set of H .

Proof. Let us prove the independence first. Assume by contradiction that the set of active vertices of some steady state map $C(H)$ is not an independent set of vertices of the graph H . Then there is a pair of adjacent vertices in H , which are simultaneously active under this steady state map M . But by lemma 1 there are no two adjacent active vertices under steady state map. This contradiction proves the independence of the set of active vertices under the steady state map. Now we consider maximality of the set of active vertices under a steady state map. Assume that there is a some inactive vertex which not adjacent to any active vertex. So that it may be added to this set to form the bigger independent set. But if such vertex exist then the map is not steady by lemma 2.

Lemma 4. Any maximal independent set of some simple graph H forms the full set of active vertices of some steady state map of the combinatorial net that derived from the H .

Proof. The desired map M is constructed as follows. Let all vertices from the maximal independent set be active under M , but another vertices let be inactive under M . Evidently, this map M is the steady state map.

Theorem about steady states of the network of bi-stable switches .
Let $C(H)$ be the combinatorial net derived from simple graph H , then the map M is steady state if and only if the set of all active vertices under map M is maximal independent set of vertices of the graph H .

Proof. Lemmas 3 and 4 prove the theorem.

6 Conclusions

A similar approach to construct Markov chains for interaction graphs was developed in our earlier works for neural and gene regulatory networks [38, 39, 40, 41, 42]. Both approaches can be used to construct Markov chains of gene regulatory networks. Systems of mutually repressive elements are ubiquitous in nature. The network of bi-stable switches can be used to create models of their stable states and the self-evolution of such systems toward a stable states.

References

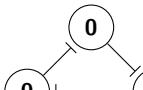
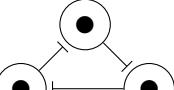
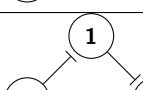
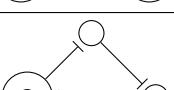
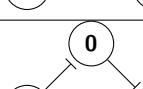
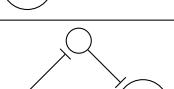
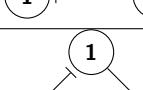
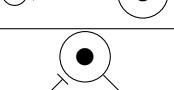
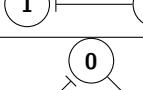
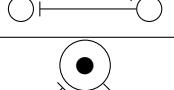
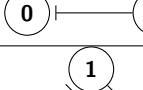
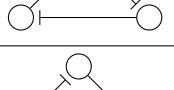
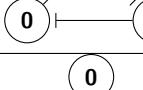
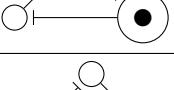
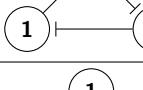
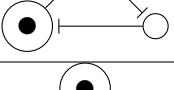
- [1] Seungchan Kim, Huai Li, Edward R. Dougherty, Nanwei Cao, Yidong Chen, Michael Bittner, and Edward B. Suh. Can markov chain models mimic biological regulation? *Journal of Biological Systems*, 10(04):337–357, 2002.
- [2] I. Shmulevich and E.R. Dougherty. From Boolean to probabilistic Boolean networks as models of genetic regulatory networks. *Proceedings of the IEEE*, 90(11):1778–1792, November 2002.
- [3] Lorenzato Almeida, Denise Regina Pechmann, and Adelmo Luis Cechin. Gene expression analysis using markov chains extracted from rnns. *CLEI Electroni Journal*, 10(2):1–10, 2007.
- [4] Jinghang Liang and Jie Han. Stochastic Boolean networks: an efficient approach to modeling gene regulatory networks. *BMC systems biology*, 6(1):113, January 2012.
- [5] Shu-Qin Zhang, Wai-Ki Ching, Yue Jiao, Ling-Yun Wu, and R.H. Chan. A simplified multivariate markov chain model for the construction and control of genetic regulatory networks. In *Bioinformatics and Biomedical Engineering, 2008. ICBBE 2008. The 2nd International Conference on*, pages 569–572, May 2008.
- [6] Stuart A. Kauffman. *The origins of order: self-organization and selection in evolution*. New York: Oxford University Press., 1993.
- [7] Xiaoning Qian and Edward R Dougherty. Intervention in gene regulatory networks via phenotypically constrained control policies based on long-run behavior. *IEEE/ACM transactions on computational biology and bioinformatics / IEEE, ACM*, 9(1):123–36, 2012.
- [8] Ritwik Layek, Aniruddha Datta, Ranadip Pal, and Edward R Dougherty. Adaptive intervention in probabilistic boolean networks. *Bioinformatics (Oxford, England)*, 25(16):2042–8, August 2009.

- [9] Aniruddha Datta, Ashish Choudhary, Michael L Bittner, and Edward R Dougherty. External control in Markovian genetic regulatory networks: the imperfect information case. *Bioinformatics (Oxford, England)*, 20(6):924–30, April 2004.
- [10] Wentao Zhao, Erchin Serpedin, and Edward R Dougherty. Identifying genes involved in cyclic processes by combining gene expression analysis and prior knowledge. *EURASIP journal on bioinformatics & systems biology*, 2009:683463, January 2009.
- [11] Mohammadmahdi R Yousefi and Edward R Dougherty. Intervention in gene regulatory networks with maximal phenotype alteration. *Bioinformatics (Oxford, England)*, 29(14):1758–67, July 2013.
- [12] Babak Faryabi, Golnaz Vahedi, Jean-Francois Chamberland, Aniruddha Datta, and Edward R Dougherty. Optimal constrained stationary intervention in gene regulatory networks. *EURASIP journal on bioinformatics & systems biology*, 2008:620767, January 2008.
- [13] Babak Faryabi, Jean-Francois Chamberland, Golnaz Vahedi, Aniruddha Datta, and Edward R. Dougherty. Optimal intervention in semi-Markov-based asynchronous genetic regulatory networks. *2008 American Control Conference*, 2008:1388–1393, June 2008.
- [14] Golnaz Vahedi, Babak Faryabi, Jean-Francois Chamberland, Aniruddha Datta, and Edward R Dougherty. Optimal intervention strategies for cyclic therapeutic methods. *IEEE transactions on bio-medical engineering*, 56(2):281–91, February 2009.
- [15] Babak Faryabi, Golnaz Vahedi, Aniruddha Datta, Jean-Francois Chamberland, and Edward R Dougherty. Recent advances in intervention in markovian regulatory networks. *Current genomics*, 10(7):463–77, November 2009.
- [16] Ranadip Pal, Sonal Bhattacharya, and Mehmet Umut Caglar. Robust approaches for genetic regulatory network modeling and intervention: A review of recent advances. *IEEE Signal Process. Mag.*, 29(1):66–76, 2012.
- [17] J J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982.
- [18] A. V. M. Herz and C. M. Marcus. Distributed dynamics in neural networks. *Physical Review E*, 47(3):2155–2161, 1993.
- [19] Xin Wang, Arun K. Jagota, Fernanda Botelho, and Max H. Garzon. Absence of Cycles in Symmetric Neural Networks. *Neural Computation*, 10:1235–1249, 1998.
- [20] T. Grossman. Applying the INN model to the MaxClique problem. In D. S. Johnson and M. A. Trick, editors, *DIMACS Series: Second DIMACS Challenge*, pages 125–145. American Mathematical Society, Providence, RI : Baltimore, MD, 1996.

- [21] A. Jagota. *The Hopfield-style network as a maximal-clique graph machine*. Buffalo TR. State University of New York at Buffalo, Department of Computer Science, 1990.
- [22] Tal Grossman and A. Jagota. On the equivalence of two Hopfield-type networks. In *International Symposium on Neural Networks*, 1993.
- [23] Arun Jagota. Approximating maximum clique with a Hopfield network. *IEEE Transactions on Neural Networks*, 6:724–735, 1995.
- [24] Arun Jagota. Hopfield neural networks and self-stabilization. *Chicago Journal of Theoretical Computer Science*, 1999(6):1–24, August 1999.
- [25] Nigel P. Duffy and Arun K. Jagota. Generalized Connectionist Associative Memory. In *International Joint Conference on Artificial Intelligence*, pages 833–839, 1999.
- [26] Arun Jagota and Xin Wang. *Oscillations in Neural Systems*, chapter Oscillations in Discrete and Continuous Hopfield Networks, pages 369 – 388. Lawrence Erlbaum Associates, Mahwah, New Jersey, 2000.
- [27] Uri Alon. Network motifs: theory and experimental approaches. *Nature reviews. Genetics*, 8(6):450–61, June 2007.
- [28] John Hertz, Anders Krogh, and Richard G. Palmer. *Introduction to the Theory of Neural Computation*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1991.
- [29] Daniel Madar, Erez Dekel, Anat Bren, and Uri Alon. Negative auto-regulation increases the input dynamic-range of the arabinose system of Escherichia coli. *BMC systems biology*, 5(1):111, January 2011.
- [30] Pablo Navarro, Nicola Festuccia, Douglas Colby, Alessia Gagliardi, Nicholas P Mullin, Wensheng Zhang, Violetta Karwacki-Neisius, Rodrigo Osorno, David Kelly, Morag Robertson, and Ian Chambers. OCT4/SOX2-independent Nanog autorepression modulates heterogeneous Nanog gene expression in mouse ES cells. *The EMBO journal*, 31(24):4547–62, December 2012.
- [31] T S Gardner, C R Cantor, and J J Collins. Construction of a genetic toggle switch in Escherichia coli. *Nature*, 403(6767):339–42, January 2000.
- [32] Jongmin Kim, Kristin S White, and Erik Winfree. Construction of an in vitro bistable circuit from synthetic transcriptional switches. *Molecular systems biology*, 2:68, January 2006.
- [33] Lucia Marucci, David a W Barton, Irene Cantone, Maria Aurelia Ricci, Maria Pia Cosma, Stefania Santini, Diego di Bernardo, and Mario di Bernardo. How to turn a genetic circuit into a synthetic tunable oscillator, or a bistable switch. *PloS one*, 4(12):e8083, January 2009.
- [34] I. Shmulevich and E.R. Dougherty. *Probabilistic Boolean Networks: The Modeling and Control of Gene Regulatory Networks*. SIAM e-books. Society for Industrial and Applied Mathematics, 2010.

- [35] M B Elowitz and S Leibler. A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767):335–8, January 2000.
- [36] Y. Shrivastava, S. Dasgupta, and S.M. Reddy. Neural network solutions to a graph theoretic problem. In *Circuits and Systems, 1990., IEEE International Symposium on*, pages 2528–2531 vol.4, May 1990.
- [37] Arun Jagota, Giri Narasimhan, and Ľuboňir Šoltés. A Generalization of maximal independent sets. *Discrete Applied Mathematics*, 109(3):223–235, May 2001.
- [38] В.П. Скорняков and А.В. Шурыгина. Применение метода Магу для перечисления устойчивых мозаик нейронной активности в комбинаторной модели элементарного возбудительно-тормозного дивергентного нейронного ансамбля. In *Материалы XVI Международной конференции по нейрокибернетике*, pages 199–202. Ростов-на-Дону. Издательство ЮФУ, 2012.
- [39] Skornyakova Maria V. Combinatorial net as a formalism for phase net forming model. In *FENS Abstr.*, volume 4, page 224.2, 2008.
- [40] Скорняков В.П. Разработка программы для дискретного математического моделирования сетей экспрессии генов, в частности, генетических осцилляторов. In *Материалы Всероссийской Молодежной Научной школы "Эффективная работа над диссертацией"*, pages 190–191. Ростов, 2012.
- [41] Скорняков П.В., Скорняков В.П, Скрякова М.В., Шурыгина А.В., and Сухов А.Г. Программный комплекс combinatorial.net для исследования сетей регулирования экспрессии генов с помощью комбинаторных моделей. In *Материалы конференции V Международная научно-практическая конференция "Актуальные проблемы биологии, нанотехнологий и медицины" 3-5 Октября 2013г.*, pages 42–44. Ростов-на-Дону. Издательство ЮФУ, 2013.
- [42] Шурыгина А.В. Вероятностная комбинаторная сеть для моделирования динамики экспрессии генов. In *Материалы Всероссийской молодежной научной школы "Эффективная работа над диссертацией"*, pages 231–233. Ростов-на-Дону: Издательство Южного федерального университета, 2012.

Table 1: Decimal, binary, and graphical representations of the state space of the Elowitz repressilator

Dec	Binary	Graphic	Non steady vertices	Direct reachable states
0	000			0,1,2,3,4,5,6,7
1	001			1, 3
2	010			2, 6
3	011			3, 2
4	100			4, 5
5	101			5, 1
6	110			6, 4
7	111			0, 1, 2, 3, 4, 5, 6, 7